

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 05-204744

(43)Date of publication of application : 13.08.1993

(51)Int.Cl.

G06F 12/02  
 G06F 3/153  
 G06F 15/66  
 // G09G 5/36  
 H04N 1/387

(21)Application number : 04-034060

(71)Applicant : NIPPON TELEGR &amp; TELEPH CORP &lt;NTT&gt;

(22)Date of filing : 27.01.1992

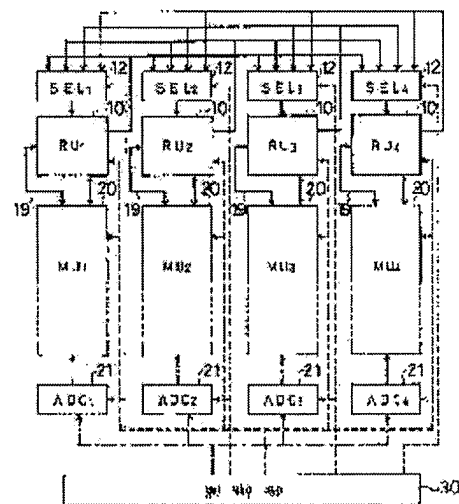
(72)Inventor : KONDO TOSHIO

## (54) TWO-DIMENSIONALLY ARRANGED DATA ACCESS DEVICE

(57)Abstract:

PURPOSE: To realize the more compact two-dimensionally arranged data access device without lowering high-speed rotation.

CONSTITUTION: This device is composed of plural memory units 20, a address translator 21 to generate addresses mutually different and a common address for all respective memory units, a rotary unit 10 equipped with a function to rotate arranged data for the unit of a block by 90° while enabling mutual access between the memory units, and a means to transfer the arranged data of the unit of the block held in the respective rotary units to the memory unit different from an access source. Thus, assuming that the rotation by 90° is parallelly executed for the unit of the block, the rearrangement of data and the address translation of the unit memory can be wholly executed for the unit of the block. Therefore, two-dimensional access is realized for the unit of 90° rotation, data arranged in the shape of laterally long slips (corresponding to rows) or data arranged in the shape of longitudinally long slips (corresponding to columns).



## LEGAL STATUS

[Date of request for examination] 14.01.1998

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 2887369

[Date of registration] 19.02.1999

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平5-204744

(43)公開日 平成5年(1993)8月13日

(51)Int.Cl. <sup>5</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/02	5 2 0	8841-5B		
3/153	3 2 0 B	9188-5B		
15/66	3 5 0	8420-5L		
// G 0 9 G 5/36		9177-5G		
H 0 4 N 1/387		4226-5C		

審査請求 未請求 請求項の数4(全 25 頁)

(21)出願番号 特願平4-34060

(22)出願日 平成4年(1992)1月27日

(71)出願人 000004226

日本電信電話株式会社

東京都千代田区内幸町一丁目1番6号

(72)発明者 近藤 利夫

東京都千代田区内幸町1丁目1番6号 日

本電信電話株式会社内

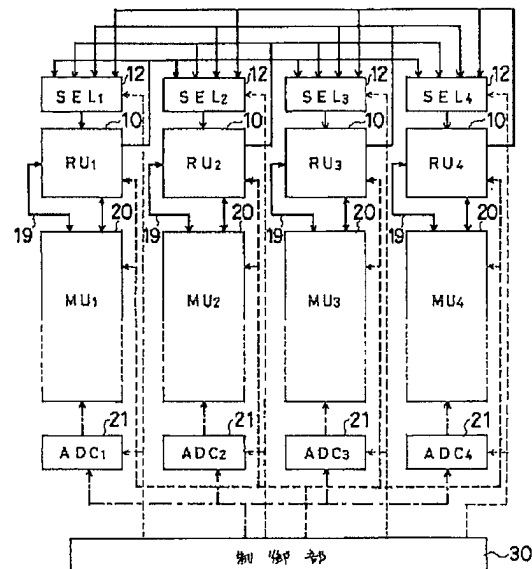
(74)代理人 弁理士 山川 政樹

(54)【発明の名称】 2次元配列データアクセス装置

(57)【要約】

【目的】 2次元配列データアクセス装置において、回転高速を低下させることなく、よりコンパクトに実現するようにする。

【構成】 複数のメモリユニット20と、その各メモリユニットに全体に共通のアドレスと互いに異なるアドレスとを生成するアドレス変換器21と、メモリユニットとの間で相互にアクセスが可能で、ブロック単位の配列データの90度回転機能を有する回転ユニット10と、各回転ユニットが保持するブロック単位の配列データをアクセス元とは異なるメモリユニットへ転送する手段から構成する。これにより、ブロック単位の90度回転の並列実行を前提に、データの並べ替え、メモリユニットのアドレスの変換が、すべてブロック単位で可能になる。そのため、90度回転、或は横長の短冊状配列データ(行に対応)、縦長の短冊状配列データ(列に対応)を単位とする2次元アクセスを実現する。



## 【特許請求の範囲】

【請求項1】 全体に共通のアドレスをアクセスする機能とユニットごとに異なるアドレスをアクセスする機能とを兼備する複数のメモリユニットと、前記各メモリユニットから読み出したブロック単位の配列データを同時に90度回転する手段と、そのブロック単位の配列データをアクセス元のメモリユニットとは異なるメモリユニットに転送する手段とを持つことを特徴とする2次元配列データアクセス装置。

【請求項2】 請求項1記載の2次元配列データアクセス装置において、90度回転する手段として、ブロック単位の配列データから、列単位、行単位のいずれでもアクセス可能な記憶回路の2次元配列からなる回転ユニットを用いることを特徴とする2次元配列データアクセス装置。

【請求項3】 請求項1記載の2次元配列データアクセス装置において、メモリユニットごとの個別のアドレス信号を、メモリユニットからアクセスする2次元配列データの回転手段に、メモリユニットからデータを読み込む際に、同時にその回転手段より出力されるデータから生成することを特徴とする2次元配列データアクセス装置。

【請求項4】 請求項2記載の2次元配列データアクセス装置において、ブロック単位の配列データを、アクセス元のメモリユニットとは異なるメモリユニットへ転送する手段として、記憶回路の2次元配列からなる回転ユニットをリング状に接続して構成することを特徴とする2次元配列データアクセス装置。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、高速でかつ小型経済的な画像処理装置、パターン認識装置等の実現に必要な2次元配列データアクセス装置に関するものである。

## 【0002】

【従来の技術】 画像処理、パターン認識、行列計算等では、2次元配列データに対し、行単位のアクセスだけでなく、列単位のアクセスがしばしば必要となる。従って、これらの分野への適用を目指す処理装置の性能向上には、行、列両方向のアクセス機能、あるいは列を行として（あるいは行を列として）アクセスするための行、列間の変換、すなわち2次元配列データの90度回転の高速化が必須といえる。従来、行、列両方向のアクセス機構としては、2次元アクセスメモリ（文献：元岡他「二次元記憶を用いた連想処理システム」信学技報EC76-80）が知られている。また、高速な90度回転機構としては、行単位で順次入力した2次元配列データを列単位で順次出力する機能（あるいは列単位で入力した2次元データを行単位で出力する機能）を有する直並列変換形の回転器（文献：特願昭57-121753）が用いられている。以下、本発明と関係の深い、これらについて

説明する。

【0003】 はじめに、2次元アクセスメモリについて説明する。図20は、2次元アクセスメモリのブロック構成図である。この2次元アクセスメモリでは、各メモリユニット20の要素データの並びをデータ並べ替えネットワーク5により適当に置換するとともに、各メモリのアドレスをアドレス変換器（ADC）21で適当に変換することにより任意の列あるいは行を1マシンスイクルでアクセスできる。以下、その動作を、アドレス変換器21が加算型でデータ並べ替えネットワーク5がパレルシフトで構成される2次元アクセスメモリに、図20に示すように1行ごとに要素1個分ずつずらして格納される2次元配列データ（ $D_{00} \sim D_{77}$ ）から、行あるいは列をアクセスする場合（要素単位のスキュードアレイ〔SkewedArray〕方式）について説明する。

【0004】 図20の2次元配列データの並びから明らかなように、各メモリユニット（MU）20の配列全体に共通のアドレスADを変換なしで与えれば、メモリユニット内の上から（ $AD+1$ ）行目の行データがアクセスでき、1番目のメモリユニット20に（ $AD+1$ ）のアドレスが行くようにアドレス変換器21を加算器として動作させれば、同図における右下がりの斜め方向の並びの一つ、すなわち（ $9-AD$ ）列目がアクセスできることがわかる。但し、このままではアクセスされるデータは巡回的にシフトされた形になっているのでデータ並べ替えネットワーク5で正規の形に戻してやる必要がある。ここで、9列目以降は、8で割った余りの列目を意味している。なお、図20中太い点鎖線はそれぞれ共通のアドレス、個別のアドレスの信号線を示している。

【0005】 次に、直並列変換形回転器を用いた90度回転機構について説明する。図21は、その回転機構の一例を示しており、2次元配列データの記憶手段であるアクセス幅がWのメモリ2、2次元配列データの行、列を変換する手段であるサイズが $W \times W$ の直並列変換型回転器1等からなる。その90度回転の動作は、 $W \times W$ の2次元配列データをW幅の行単位でメモリ2から読み出し、それを、左側から列として、順次、回転器1に入力した後、今度は下側からW幅の行単位に順次取り出して、メモリ2に書き込むことである。同図中19はメモリ、回転器間バスを示している。

【0006】 この動作から明らかなように、 $W \times W$ の2次元配列データの回転の所要マシンスイクル数 $S_v$ は、メモリから回転器への読み出しにWマシンスイクル、回転器からメモリへの書き込みにWマシンスイクルで、合わせて

$$S_v = 2W \quad \dots\dots(1)$$

となる。但し、W幅の行単位のメモリアクセス、回転器1に対する入出力は、いずれも1マシンスイクルで実現できるものとしている。これにより、メモリ2内に格納されている $A \times A$ （AはWの整数倍）の大規模な2次元

3

配列データを、 $W \times W$  単位の回転を繰り返して実現する場合、全体の所要マシンサイクル数  $S_1$  は

$$S_1 = (A^2 / W^2) \times 2W \\ = 2A^2 / W \quad \dots \dots (2)$$

となり、回転の処理速度は  $W$  に比例することがわかる。

【0007】

【発明が解決しようとする課題】しかし、これらのいずれも、列あるいは行のアクセス幅を広げて速度向上を図ろうとすると、ハードウェア規模が著しく増大する欠点がある。例えば、2次元アクセスメモリでは、図20に示すブロック構成から明かなように、行あるいは列の要素ごとに対応するメモリユニット20とアドレス変換器21を持たねばならないのに加え、データ並べ替え用ネットワーク5として、ハードウェア規模が行あるいは列のアクセス幅に比例する以上の割合で増大するパレルシフタ、シャッフルネットワーク等を必要とする。また、直並列変換型の回転器1を用いる90度回転機構も、図21から明かなように、行あるいは列のアクセス幅  $W$  の2乗に比例して、ハードウェア規模が著しく増大する。

【0008】本発明は以上の点に鑑み、上記のような課題を解決するためになされたもので、その目的は、画像やパターンなどの2次元配列データを高速に処理する装置を小型経済的に実現するため、回転速度を低下させることなくハードウェア規模をよりコンパクトに実現可能にした2次元配列データアクセス装置を提供することにある。

【0009】

【課題を解決するための手段】上記の目的を達成するため、本発明では、従来の2次元アクセスメモリのようにデータの並べ替え、メモリユニットのアドレスの変換を要素単位で行うのではなく、ブロック単位の90度回転の並列実行を前提に、データの並べ替え、メモリユニットのアドレスの変換を、すべてブロック単位で行うことによって、90度回転、あるいは横長の短冊状配列データ（行に対応）、縦長の短冊状配列データ（列に対応）を単位とする2次元アクセスを実現する。

【0010】具体的には、全体に共通なアドレスと互いに異なるアドレスとをアクセスできるメモリユニットのアレイと、その各メモリユニットからアクセスされるブロック単位の配列データを同時に90度回転することのできる90度回転ユニットのアレイと、各回転ユニットが保持するブロック単位の配列データを、アクセス元とは異なるメモリユニットへ転送する手段とで、2次元配列データアクセス装置を構成するものである。

【0011】

【作用】本発明においては、各90度回転ユニットが、対応する各メモリユニットの所定（2次元配列データを回転するのか、縦長、横長の短冊状配列データをアクセスするのか）によって決まる。配列全体に共通の場合とメ

4

モリユニットごとに異なる場合がある。）のアドレスからブロック単位の配列データを読み出し、続いてこの各ブロック単位の配列データを、必要に応じて90度回転ユニット配列で並列に90度回転して、さらにその各ブロック単位の配列データを転送手段を介して、やはり所定のメモリユニットの互いに異なるアドレスに書込むことにより、2次元配列データの90度回転、あるいは2次元配列データに対する横長の短冊状配列データおよび縦長の短冊状配列データのアクセスが可能となる。これによって、回転の処理速度を低下させることなく、データ並べ替え用ネットワークの単純化、アドレス変換器の低減等によるハードウェア規模の低減がはかれる。

【0012】

【実施例】

実施例1

図1は本発明の第1の実施例であり、この実施例は、図1に示すように、回転ユニット（ $RU_1 \sim RU_4$ ）10、セクタ（ $SEL_1 \sim SEL_4$ ）12、アクセス幅が  $W/N$ （本実施例では  $N=4$ ）のメモリユニット（ $MU_1 \sim MU_4$ ）20、アドレス変換器（ $ADC_1 \sim ADC_4$ ）21等の1次元配列、および制御部30から構成される。ここで、符号19は各メモリユニット、回転ユニット間バスであり、太い実線はアクセス幅が  $W/4$  のデータ転送用信号線、太い破線は読み、書き、転送等の制御のための共通の制御信号線、細い破線は個別の制御信号線、太い一点鎖線はアドレス信号線である。

【0013】また、図2は回転ユニット（ $RU_1 \sim RU_4$ ）10の構成を示しており、記憶回路（MC）13の2次元配列、行単位に設けている入力バス14、出力バス15、列単位に設けている出力バス16、セクタ（SEL）17等からなる。なお、本実施例の2次元配列データアクセス装置を外部からアクセスするためのメモリユニット配列あるいは回転ユニット配列に対するバスは簡単のため図示していない。以下、本実施例により、図3に示す2次元配列データを90度回転する場合について説明する。

【0014】図3の回転対象の2次元配列データは、ブロック単位の配列データのA～Pから構成されており、メモリユニット20の配列に、図4に示すようにA、E、I、Mの先頭の行からD、H、L、Pの末尾の行までが、メモリユニット20の配列のAD0～（AD0+W-1）番地に格納されているものとする。この2次元配列データを次のような手順で処理することにより、下側のAD1～（AD1+W-1）番地の格納エリアに回転後の2次元配列データを順次形成することで、90度回転を実現する。

【0015】ステップ1

A、H、K、Nの各ブロック単位の配列データを、図5に示すように、90度回転を行いながら回転ユニット10の配列に読み出す。この処理は、各メモリユニット2

5

0から順次読み出す $W/4$ 幅の行単位のデータを、各回転ユニット10で最右列から左側に向かって順次書込むことで行う。この際、メモリユニット(MU)20に与えるアドレスは、制御部30で順次生成する $AD0 \sim *$

$$ADD_i = AD0 + \text{mod}((ADD - AD0 + W - (i-1)W/4)/W) \dots \dots (3)$$

となるように変換することで生成する。また、回転ユニット10内のセクタ(SEL)17はメモリユニット20からの入力を選ばれるように設定する。さらに、各回転ユニット10への書込みは、2次元配列の記憶回路(MC)13に対する行方向の入力バス14からの書込みを、最右列から最左列に向かって列単位で順次活性化することで行う。

#### 【0016】ステップ2

回転ユニット(RU)10内のブロック単位の配列データを図6に示すように左側に回転ユニット1個分シフトする。これは、 $i$ 番目のセクタ12つまり $SEL_i$ を、 $i+1$ 番目の回転ユニット $RU_{i+1}$ からの入力を選ばれるように設定し、各回転ユニット10内のセクタ17をセクタ12つまり $SEL_i$ からの入力を選ばれるように設定するとともに、回転ユニット10内の記憶回路(MC)13の行方向バス15に対する出力、およ

$$ADD_i = AD1 + \text{mod}((ADD - AD1 + i \times W/4)/W) \dots (4)$$

となるように変換することで生成する。

【0018】以上のステップ1からステップ3の一連のサイクルを、次のサイクルに入るごとに制御部30から生成するアドレス $ADD$ を $W/4$ だけ大きくするとともに、ステップ2における左方向へのシフト量を、セクタ12のセクタの設定を切り換えることで、回転ユニット1個分ずつ大きくしながら、さらに3回繰り返すと、{B, E, L, O}、{C, F, I, P}、{D, G, J, M}のブロック単位の配列データの各組が順次変換され、図8に示すように、回転後の格納エリアに90度回転結果が得られる。

【0019】続いて、本実施例による縦長の短冊状配列データ、横長の短冊状配列データを単位とする2次元アクセスについて説明する。アクセス対象の2次元配列データ(図3に示すものと同一とする)が、あらかじめ、図9に示すように2次元配列データが $(W/4) \times (W/4)$ のブロックを単位とするスキュードアレイ形式で格納されているものとする。

【0020】この図9から容易にわかるように、従来の2次元アクセスメモリの行に対応する横長の短冊状配列★

$$ADD_i = AD0 + \text{mod}(ADD - AD0 + W - (i-1)W/4)/W) \dots \dots (5)$$

により、各メモリユニット20から順次読み出される $(W/4)$ 幅の行単位のデータを、回転ユニット10の列として右側から順に格納する(図12参照)。

【0022】続いて、回転ユニット10の配列上のデータを、回転ユニット10の所定の個数分シフトすること

6

\* ( $AD0 + W/4 - 1$ )のアドレス $ADD$ を、さらにアドレス変換器21で、左から $i$ 番目のメモリユニット $MU_i$ に対するアドレス $ADD_i$ が

※び行方向バス14からの入力(バス14からの記憶回路13に対する書込み)を、各記憶回路13内で列単位に順次活性化することで行う。

#### 10 【0017】ステップ3

各回転ユニット10内のA, H, K, Nのブロック単位の配列データを、図7に示すようにメモリユニット20内の回転後の格納エリアに書込み。この処理は、回転ユニット10内のメモリユニット20に向かう列方向バス16に対する記憶回路13の出力を、行単位で、上側から下側に向かって順次活性化し、これによって読み出されるデータをメモリユニット20に順次書込むことで行う。この際、メモリユニット(MU)20に与えるアドレスは、ステップ1と同様、制御部30で順次生成する $AD1 \sim (AD1 + W/4 - 1)$ のアドレス $ADD$ を、さらにアドレス変換器21で、左から $i$ 番目のメモリユニット $MU_i$ に対するアドレス $ADD_i$ が

★データは、制御部30で生成する全体に共通のアドレスで、そのまま回転ユニット10に読み出し(図10参照)、その後、回転ユニット10間で所定量だけシフトすることで、回転ユニット10の配列上に読み出すことができる(図11参照)。行単位のデータが必要な場合には、各回転ユニット10の対応する行を活性化し、列方向のバスを介して読み出せばよい。書込みは、この逆で、対象とする横長の短冊状配列データを所定量回転ユニット10間でシフトした後、制御部30で生成する共通のアドレスでメモリユニット20に書込むことで実現される。

【0021】これに対し、列に対応する縦長の短冊状配列データの読み出しは、共通のアドレスをアドレス変換器21で個別に変換するとともに、回転をかけながら回転ユニット10に読み出すことで行う。具体的には、先の90度回転の場合同様、はじめに、制御部30より供給される $AD0 \sim (AD0 + W - 1)$ の共通のアドレス $ADD$ をアドレス変換器21で変換して生成したアドレス $ADD_i$

で、回転ユニット配列上に目的の縦長の短冊状配列データを得ることができる(図13参照)。書込みは、この読み出しの手順を全く逆に行うことで実現できる。ただし、この場合必要となるメモリユニット20に回転をかけながら書込む処理は、回転ユニット10内で記憶回路

13の行方向バス15に対する出力を列単位で活性化することで読み出される $W/4$ の幅のデータを、左側からメモリユニット20に向かうバス19を介して転送することで行う。

#### 【0023】実施例2

次に、回転ユニット10を2次元のシフトレジスタで構成する本発明の第2の実施例について説明する。この第2の実施例の構成で、第1の実施例と異なるのは、回転ユニット11のみである。そこで、図14に、その第2の実施例の回転ユニット11の構成を示す。第1の実施例の回転ユニット10との違いは、バスの代りに、各記憶回路(MC)13が隣接する記憶回路13との間で、左から右方向、および下から上方向にシフト転送用のバスを設けていることである。

【0024】動作についても、回転ユニット11に対する読み書きを、記憶回路13を列あるいは行単位で活性化して行う代りに、記憶回路13間で、保持データあるいは入力データを上あるいは右方向に順次シフトすることで行う点が、第1の実施例と異なるだけである。この第2の実施例のように回転ユニット11をシフトレジスタ形式で構成する利点は、列単位あるいは行単位の活性制御が不要となり、その分制御が単純化されることである。

#### 【0025】実施例3

図15に本発明の第3の実施例を示す。この実施例は、第1、第2の実施例のセクタ12を省略し、回転ユニット10をリング状に接続することで構成したものである。この実施例の動作は、先のステップ2のブロック単位の配列データの回転ユニット間のシフトをすべて右方向の単位シフトで行う点が、第1、第2の実施例と異なる。従って、シフト量が多い場合に単位シフトの繰り返しが必要となり、全体の回転時間が増加する欠点がある。しかし、セクタ12が不要となる分ハードウェア規模が低減されるのに加え、メモリユニット(MU<sub>i</sub>)20、回転ユニット(RU<sub>i</sub>)10、アドレス変換器(ADC<sub>i</sub>)21等を1つのモジュールとして構成する場合、モジュール間の接続線が、モジュール数N(この実施例では4)分の1に低減される利点がある。端子数の制約が厳しいLSIやボードをモジュールとする場合、この利点は、極めて有効である。

【0026】なお、単位シフトのみでブロック単位の配列データをシフトする場合、容易に理解できるように、シフト方向を双方向化することで、全体のシフト量は半減できる。また、図面による説明は割愛するが、互いに隣接するモジュールの構成ユニットである回転ユニット10とメモリユニット20との間で、双方向バスを介して、データのアクセスができるようにして、回転と単位シフトを並行して行うことで、さらにシフト量を低減することができる。

#### 【0027】実施例4

ところで、本発明をプロセッサレイ型の並列プロセッサに組込む場合、プロセッシングエレメント(PE)の演算機能を利用することで、アドレス変換器をより単純に構成することができる。図16は、この考え方に基づいた本発明の第4の実施例である。ここで、回転ユニット(RU<sub>1</sub>~RU<sub>4</sub>)18は、前述した回転ユニット10、11の中間的な構成をとっており、図17に示すように、各記憶回路(MC)13の上下方向に列単位のバスを設けると共に、左右方向にはシフト転送用の双方向のバスを設けている。但し、この図17から明らかなように、この回転ユニット18は、プロセッシングエレメントが隣接プロセッシングエレメント間のデータ移動用レジスタを内蔵する場合、省略可能である。これは、回転ユニットの機能がデータ移動用レジスタの並びでエミュレートできるからである。

【0028】PEアレイ40<sub>1</sub>~40<sub>4</sub>は、 $(W/4)$ 個のプロセッシングエレメントからなるプロセッサアレイである。このプロセッシングエレメントPE<sub>j</sub>は、回転ユニット18内のj番目の列方向バスに接続されており、PEアレイ40<sub>1</sub>~40<sub>4</sub>は、各回転ユニット18の保持データを行単位でアクセスすることができる。従って、回転ユニット18の各列は、それが列バスを介して接続されるプロセッシングエレメントPEのローカルなレジスタとして機能させることができる。また、セクタ(SEL<sub>1</sub>~SEL<sub>4</sub>)22は、各メモリユニット(MU<sub>1</sub>~MU<sub>4</sub>)20に供給するアドレスを、回転ユニット(RU<sub>1</sub>~RU<sub>4</sub>)18から出力される列単位のデータとするか、制御部30から出力される全体に共通のアドレス信号とするかを選択するためのアドレス変換器に相当するセクタである。

【0029】本実施例の動作は、回転ユニット18のメモリユニット20に対するデータのアクセスに特徴があり、その他の動作については上記実施例1及び2と同じなので説明を割愛する。すなわち、このアクセス(読み出し)は以下の手順で行う。

①あらかじめ各PEで保持しているPEアレイ40<sub>1</sub>~40<sub>4</sub>ごとのベースアドレスADb<sub>i</sub>(=AD0+mod((W-(i-1)(W/4))/W))と、やはり、あらかじめ各PEで保持しているPEのアレイ内アドレスj-1とから、ADD<sub>ij</sub>(例えば、=ADb+(j-1))を各PEで求め、結果を、各PEに接続される回転ユニット18内の記憶回路(MC)13の列に転送する(図18参照)。

【0030】②回転ユニット18で左から右に順次保持データをシフトさせ、出力のADD<sub>ij</sub>でメモリユニット20をアドレッシングしながら、これによってメモリユニット20から読み出されるデータ[ADD<sub>ij</sub>]を、同時に順次回転ユニット18の左側から入力する。この際、セクタ22を回転ユニット18からの出力ADD<sub>ij</sub>が

50 選択されるように、回転ユニット18内のセクタ17

をメモリユニット20からの入力を選択されるように、それぞれ設定しておく。

【0031】これにより②の処理を(W/4)回繰り返すことで、図19に示すように、メモリユニット20内のデータ[ADD<sub>10</sub>]～[ADD<sub>1((W/4)-1)</sub>]が、回転ユニット18に読み出され、実施例1のステップ1と等価な処理が実現される。このように、本実施例では、アドレス変換器が、プロセッサの演算機能と回転ユニットのシフト機能を利用することで、実効的にセクタ22のみで構成される。

【0032】なお、実施例1のステップ3のメモリユニット20への書き込みの処理は、回転途中のデータを、一旦、回転ユニットからプロセッシングエレメントPEに移しておき、①と同様にしてアドレスを生成しながら、列バスを介して、プロセッシングエレメントPEからメモリユニットへ直接データを転送することで行う。ここで、一旦、回転途中のデータを回転ユニットからプロセッシングエレメントPEに移すのは回転ユニットを、メモリユニットに対するアドレスデータADD<sub>11</sub>の生成用に空けるためである。

【0033】ここで、90度回転に関し、本発明の最も単純な構成の実施例である第4の実施例のモジュール数をNとする一般的な場合について、ハードウェア規模、所要マシンサイクル数等を、従来方式と比較して評価する。はじめに、ハードウェア規模について評価する。図1の構成から明らかなように、メモリ部分については(W/N)の幅のメモリユニットに分割されているだけであり、アドレス変換器についてもセクタのみで構成されるので、従来方式に対するハードウェア規模の増加は無視できる程度である。

【0034】これに対し、90度回転器の部分については、本発明では、(W/N)×(W/N)のサイズの回転ユニットがN個で構成されるので、回転ユニットとして従来例と同一のものをを用いるとすれば、そのハードウェア規模はW<sup>2</sup>/Nと従来方式W<sup>2</sup>に比べ1/Nになる。また、第4の実施例のように、プロセッサアレイ型の並列プロセッサに組込む場合は、回転ユニットの配列を各プロセッシングエレメントに内蔵する隣接プロセッシングエレメント間のデータ移動用レジスタでエミュレートすることもできるので、ハードウェア規模低減の量はさらに大きいと言える。

【0035】しかし、所要なマシンサイクル数については、若干増加する。以下、この所要マシンサイクル数を算出する。まず、ステップ1のブロック単位の配列データを、各回転ユニットに読み出す処理では、メモリユニットからの読み出しと回転ユニットへの格納が共に同一マシンサイクル内で実行できるとすると、その所要マシンサイクル数S<sub>1</sub>は、ブロック単位の配列データの構成行数に等しく、

$$S_1 = W/N \quad \dots\dots (6)$$

となる。

【0036】これに続く、ステップ2の回転ユニット間のブロック単位の配列データのシフトに要するマシンサイクル数S<sub>2</sub>は、ブロック単位の配列データのアクセス位置により異なるが、平均すると

$$S_2 = (N/4) \times W/N = W/4 \quad \dots\dots (7)$$

となる。ただし、これは、双方向のシフトおよびステップ1での回転と逆方向シフトの並列実行等により、シフト量の低減を図った場合の値である。ここで、Nは偶数、回転ユニット間の列単位のデータのシフトは、1マシンサイクルで実行できるものとしている。

【0037】さらに、これに続く、ステップ3の各メモリユニットへの書き込みの処理の所要マシンサイクル数S<sub>3</sub>は、回転ユニットからメモリへの行単位の書き込みが1マシンサイクルで実行できるとすると、

$$S_3 = W/N \quad \dots\dots (8)$$

となる。従って、W行×W列の2次元配列データ全体を、90度回転するための所要サイクル数S<sub>T</sub>は、

$$S_T = N \times (S_1 + S_2 + S_3) = 2W + (N \times W/4) \quad \dots\dots (9)$$

となる。

【0038】このトータルの所要マシンサイクル数S<sub>T</sub>は、W×Wの従来の90度回転器に比べると、第2項のステップ2の回転ユニット間のブロック単位の配列データのシフトに起因する分が増加している。しかし、ハードウェア規模、すなわちコストを考慮した回転器部分の性能/コストでは、回転器のハードウェア規模が1/Nに低減されることから、かなり改善される。例えば、N=8の場合では、(9)式より、性能/コストは従来の4倍にもなる。

【0039】一方、本発明の2次元アクセスメモリとしての従来装置との比較、評価は、本発明の2次元アクセスが、短冊状の2次元配列データに対して実現される点で機能が若干低下することから、数値化することが難しいので割愛する。しかし、プロセッサアレイ型の並列プロセッサに組込む場合には、アドレス変換器が実効的に不要になること、回転ユニットの配列がプロセッシングエレメント内のデータ移動用レジスタを流用することで構成できること等と、並列プロセッサのハードウェア規模をほとんど増加させることなく、従来の2次元アクセスに近い機能を実現できる点で、2次元アクセスを必要とする並列プロセッサの小型化、経済化に大きく寄与するものと考えられる。

【0040】なお、以上の説明では、メモリのアクセス幅、行列変換型回転ユニットのサイズ等の単位については触れていないが、これは2次元配列データの要素の語長によって決まる。すなわち、2次元配列データの要素の語長が1ビット(2値データ)ならば、単位はビットとなる(この場合、例えば幅がWとは、幅がWビットの

ことである。)要素の語長が、 $w$ ならば、単位は $w$ 倍された $w$ ビットとなる(この場合、例えば幅が $W$ とは、幅が $(W \times w)$ ビットのことである。)。また、各ユニットにから読み出したブロック単位の配列データを回転ユニットで回転する場合、その回転方向は $+90$ 度、 $-90$ 度のいずれでもよく、 $90$ 度の奇数倍回転が可能である。

【0041】

【発明の効果】以上説明したように本発明は、全体に共通なアドレスと互いに異なるアドレスとをアクセスできるメモリユニットのアレイと、その各メモリユニットからアクセスされるブロック単位の配列データを同時に $90$ 度回転することのできる $90$ 度回転ユニットのアレイと、各回転ユニットが保持するブロック単位の配列データを、アクセス元とは異なるメモリユニットへ転送する手段から構成することにより、従来のように行あるいは列のアクセス幅に対し比例以上或は $2$ 乗に比例して大規模化した $2$ 次元データアクセス装置を、回転速度を低下させることなく、回転器のハードウェア規模を大きく低減できる。また、従来の $2$ 次元アクセスメモリに近い機能も提供できる。さらに、各メモリユニット、回転ユニット等の構成ユニットを同一モジュールの規則的な配列で構成できるので、速度向上のためのメモリアクセス幅拡大が容易に実現できる利点もある。従って、 $2$ 次元配列データの $90$ 度回転を多用する画像処理装置、パターン認識装置等に、本発明を適用すれば、高速化と小型、経済化の両立が可能となる。

【0042】また、本発明は、プロセッシングエレメントが隣接間のデータ移動用レジスタを内蔵するプロセッサアレイ型の並列プロセッサに、簡単に組み込むことができる。これは、データ移動用のレジスタの並びを回転ユニットに流用することにより、セクタなどのわずかなハードウェアの追加で、本発明の $2$ 次元配列データアクセス装置が実現されるからである。従って、本発明は $2$ 次元アクセスや $90$ 度回転を高速に実行する汎用的なプロセッサアレイ型並列プロセッサの小型経済化技術としても極めて有用である。

【図面の簡単な説明】

【図1】本発明の第1の実施例のブロック構成図である。

【図2】本発明の第1の実施例の回転ユニットの構成図である。

【図3】被回転イメージデータを示す図である。

【図4】本発明の第1の実施例による回転処理を説明した図である。

【図5】本発明の第1の実施例による回転処理を説明し

た図である。

【図6】本発明の第1の実施例による回転処理を説明した図である。

【図7】本発明の第1の実施例による回転処理を説明した図である。

【図8】本発明の第1の実施例による回転処理を説明した図である。

【図9】本発明の第1の実施例による $2$ 次元アクセスを説明した図である。

10 【図10】本発明の第1の実施例による $2$ 次元アクセスを説明した図である。

【図11】本発明の第1の実施例による $2$ 次元アクセスを説明した図である。

【図12】本発明の第1の実施例による $2$ 次元アクセスを説明した図である。

【図13】本発明の第1の実施例による $2$ 次元アクセスを説明した図である。

【図14】本発明の第2の実施例の回転ユニットの構成を示す図である。

20 【図15】本発明の第3の実施例のブロック構成図である。

【図16】プロセッサアレイ型並列プロセッサに組込んだ本発明の第4の実施例のブロック構成図である。

【図17】本発明の第4の実施例の回転ユニットの構成を示す図である。

【図18】本発明の第4の実施例によるメモリユニットから回転ユニットへ読み出しを説明した図である。

【図19】本発明の第4の実施例によるメモリユニットから回転ユニットへ読み出しを説明した図である。

30 【図20】従来例による $2$ 次元アクセスメモリの構成図である。

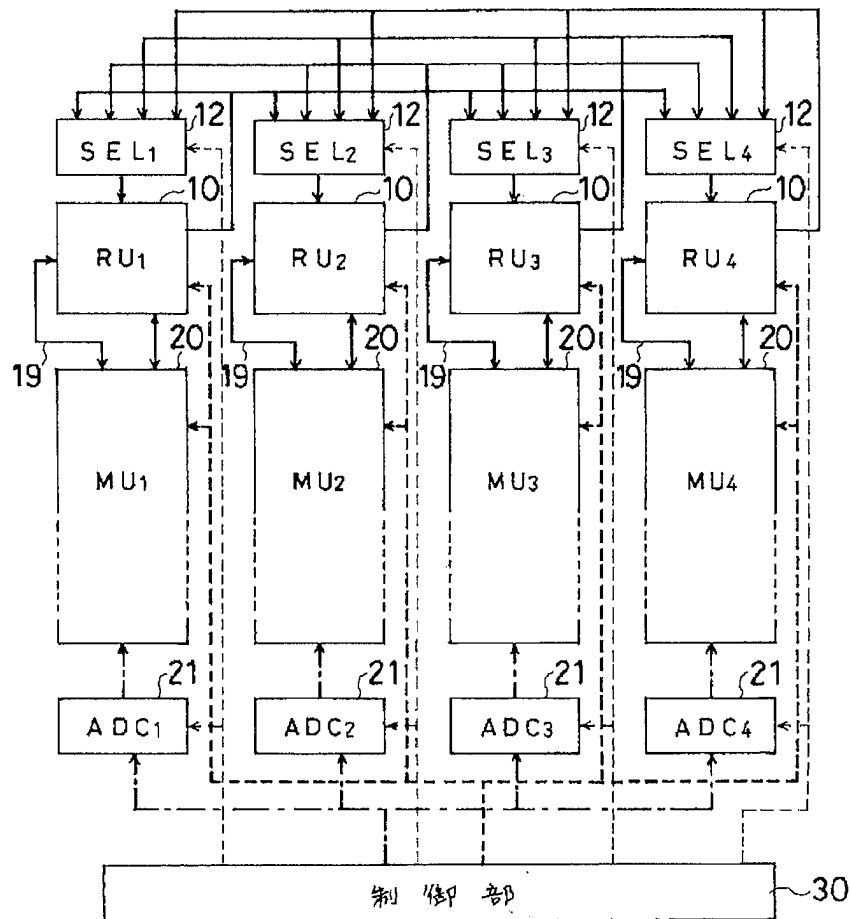
【図21】従来の直並列変換型回転器を用いた $90$ 度回転機構を説明した図である。

【符号の説明】

- 10, 11, 15, 18 回転ユニット (RU)
- 12, 17, 22 セクタ (SEL)
- 13 記憶回路 (MC)
- 14 行方向入力バス
- 15 行方向入力バス
- 40 16 列方向入力バス
- 19 メモリユニット、回転ユニット間バス
- 20 メモリユニット (MU)
- 21 アドレス変換器 (ADC)
- 30 制御部
- 40 プロセッシングエレメント配列 (PEアレイ)



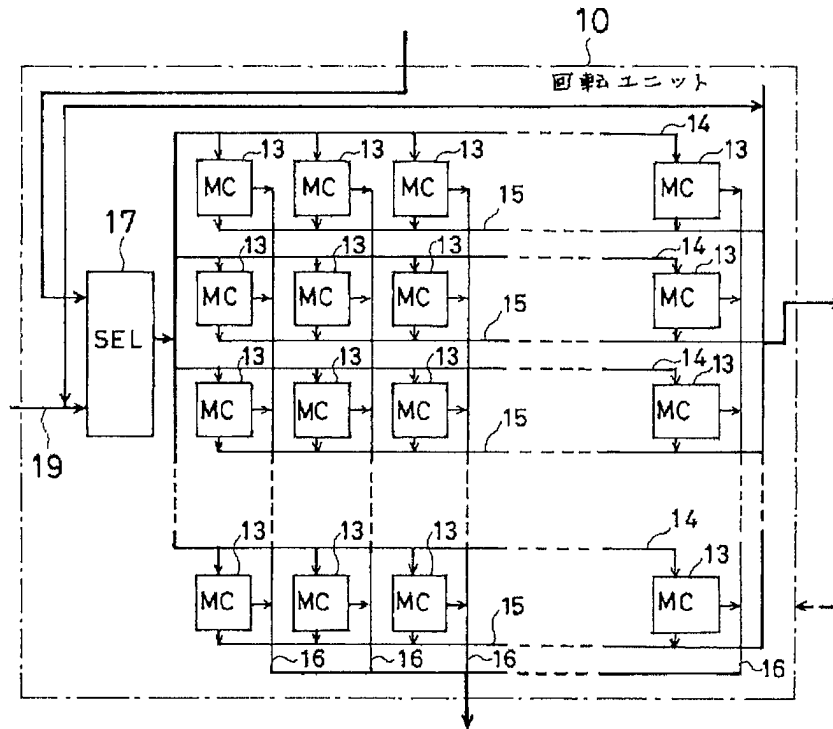
【図1】



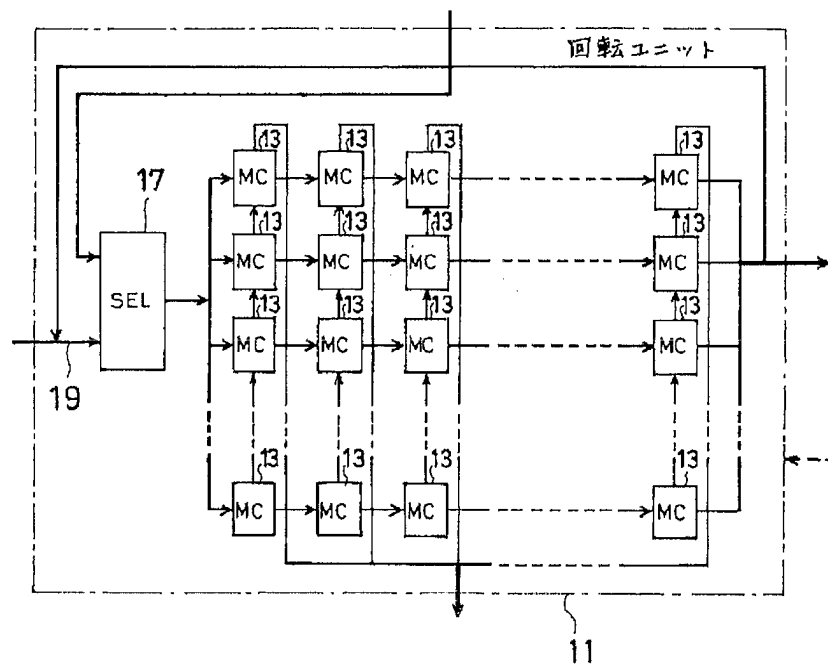
【図3】

A	E	I	M
B	F	J	N
C	G	K	O
D	H	L	P

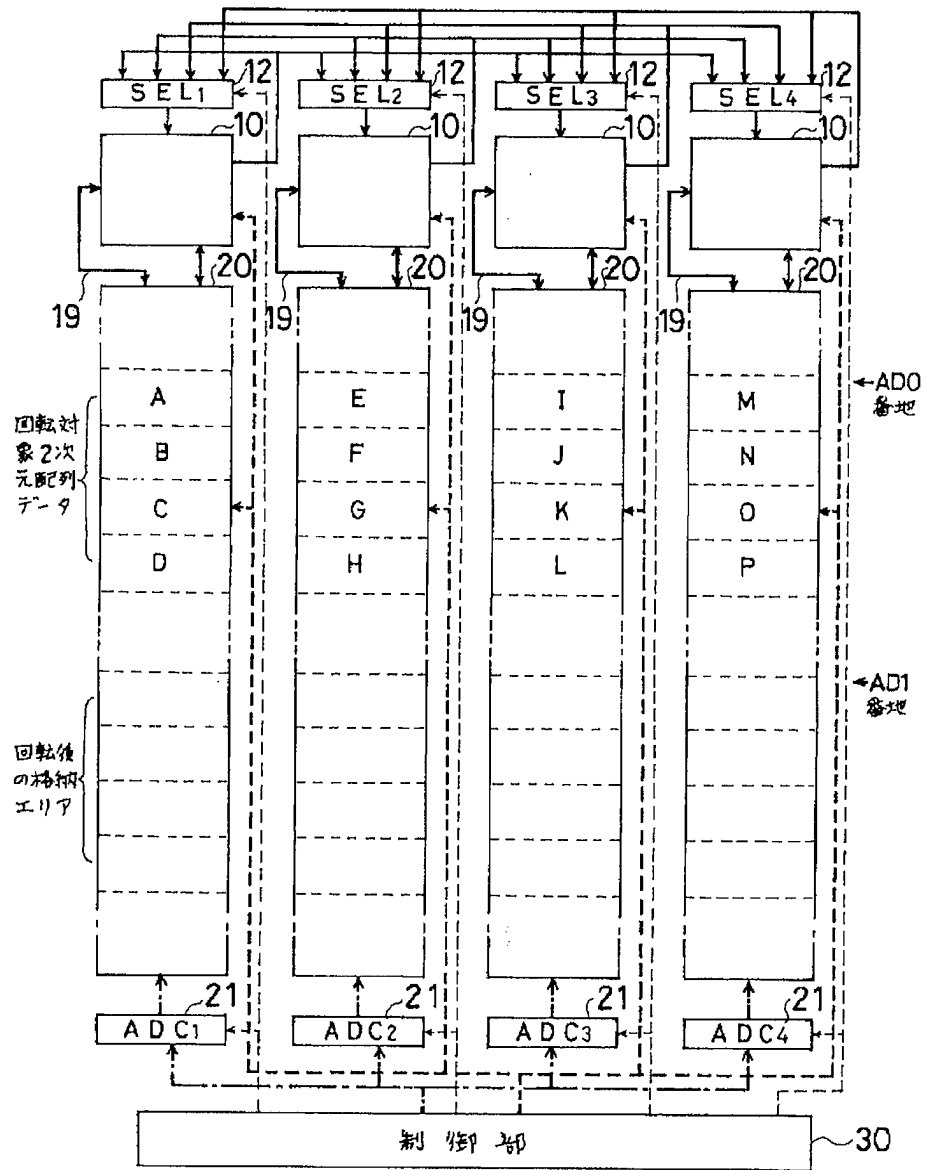
【図2】



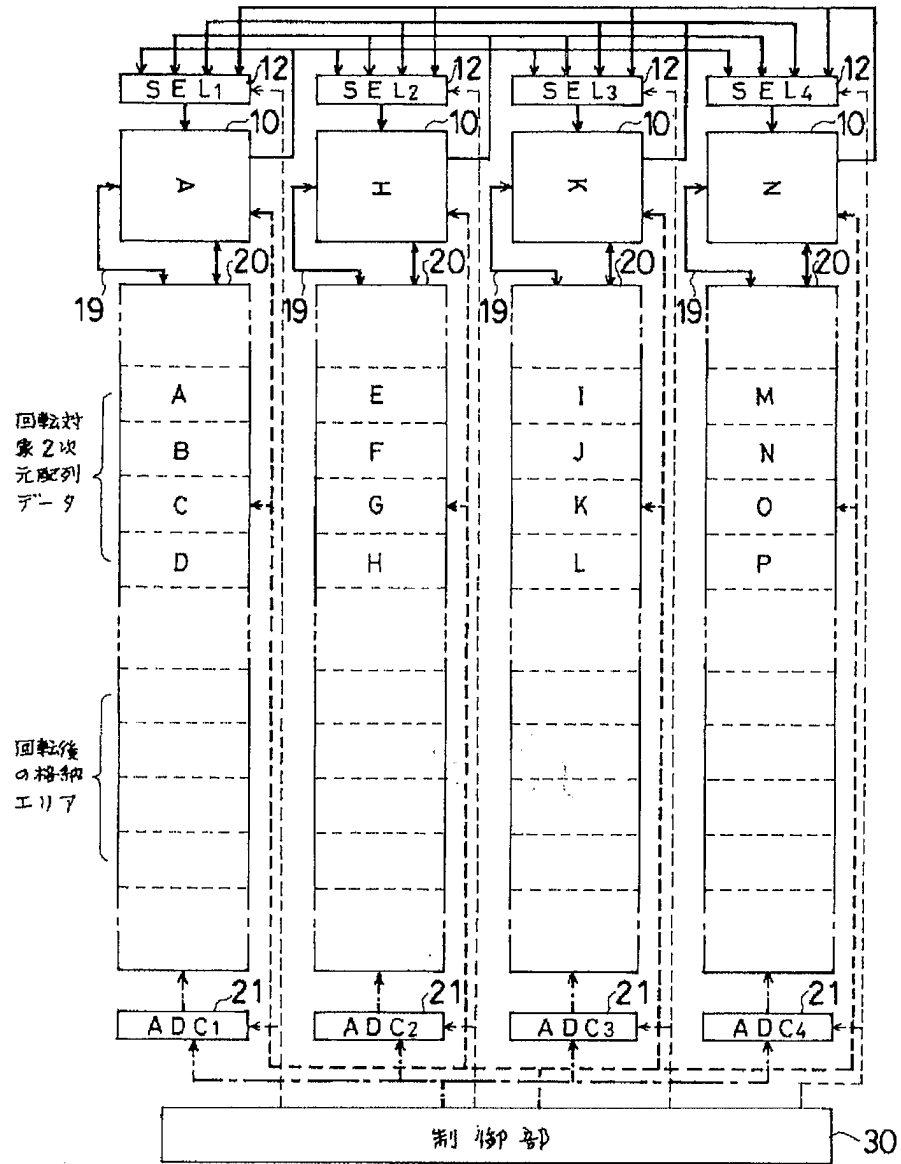
【図14】



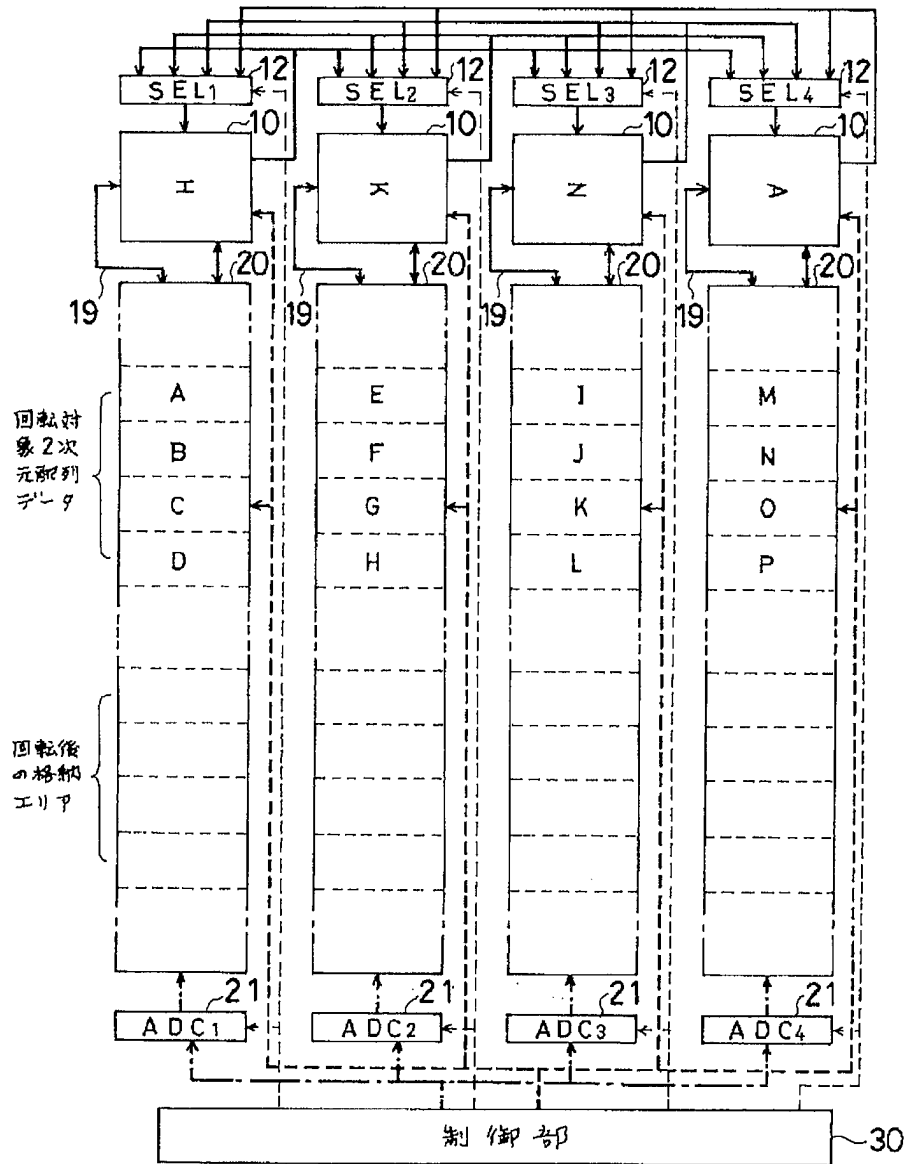
【図4】



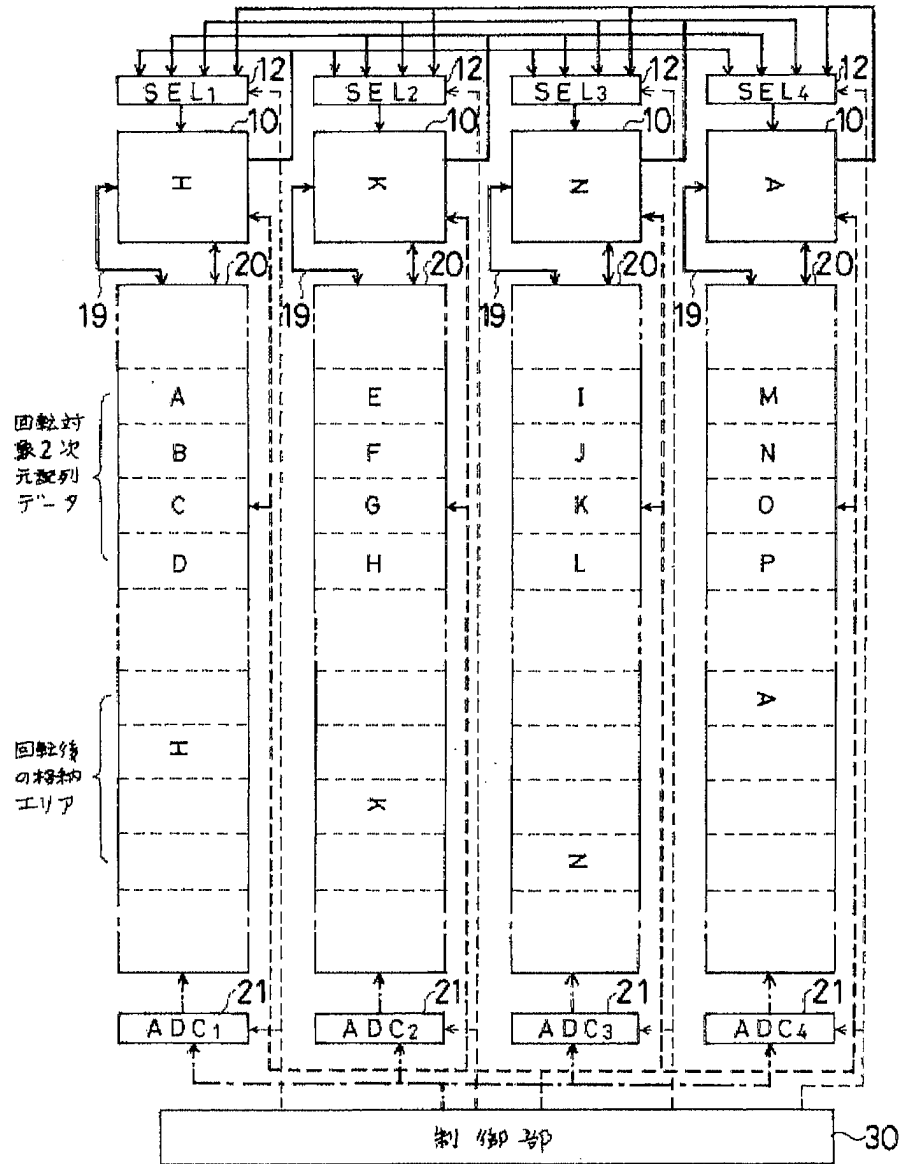
【図5】



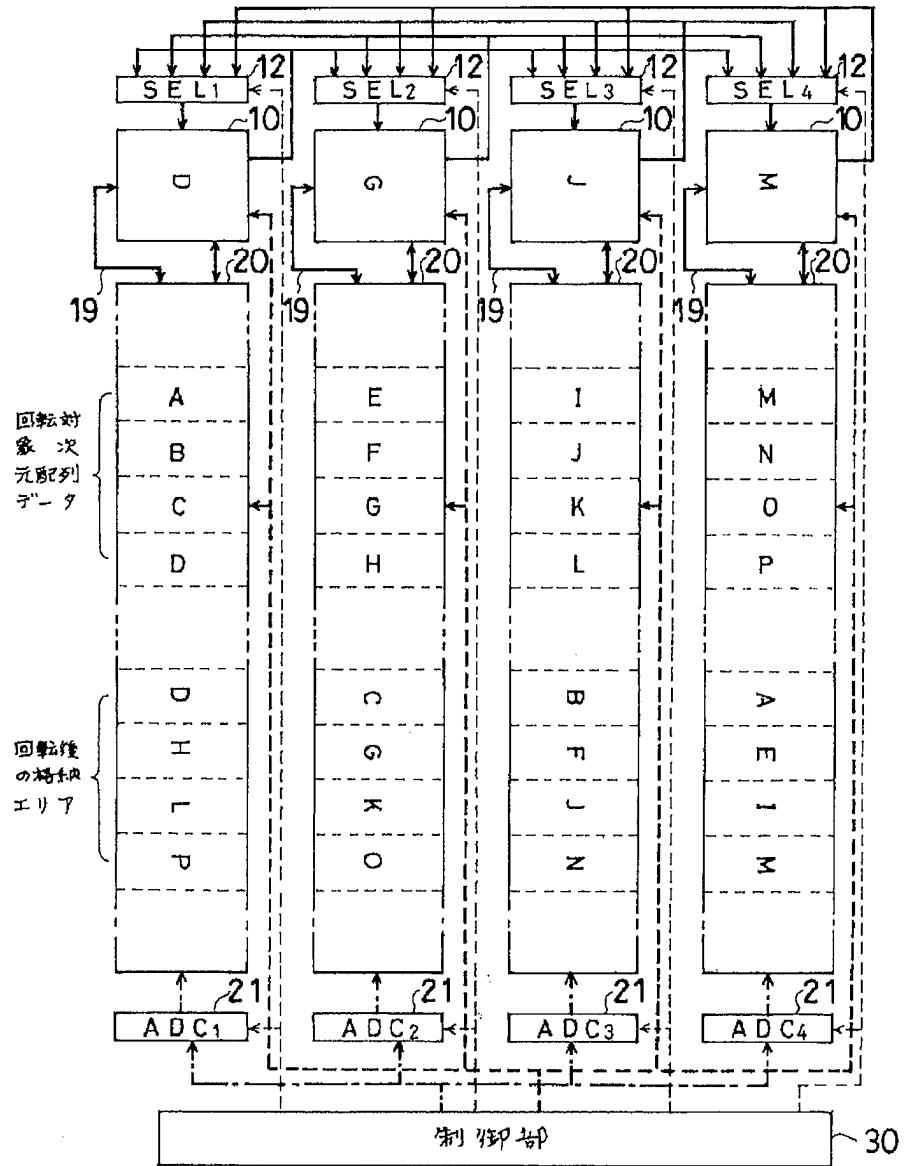
【図6】



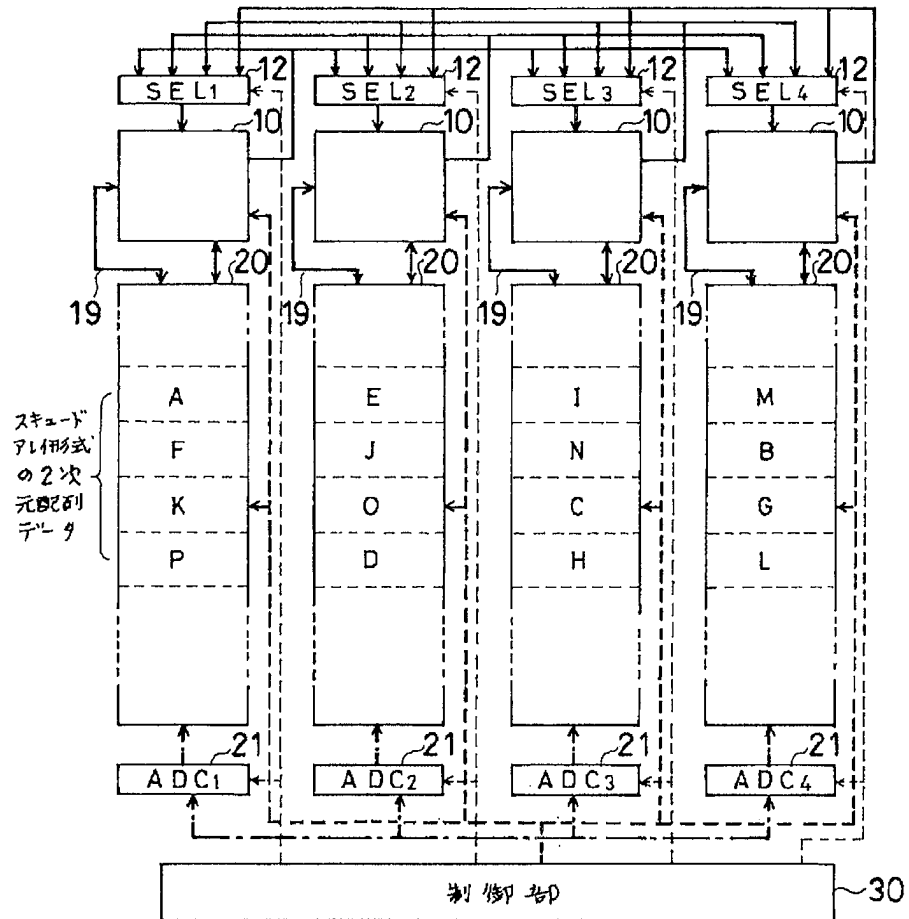
【図7】



【図8】

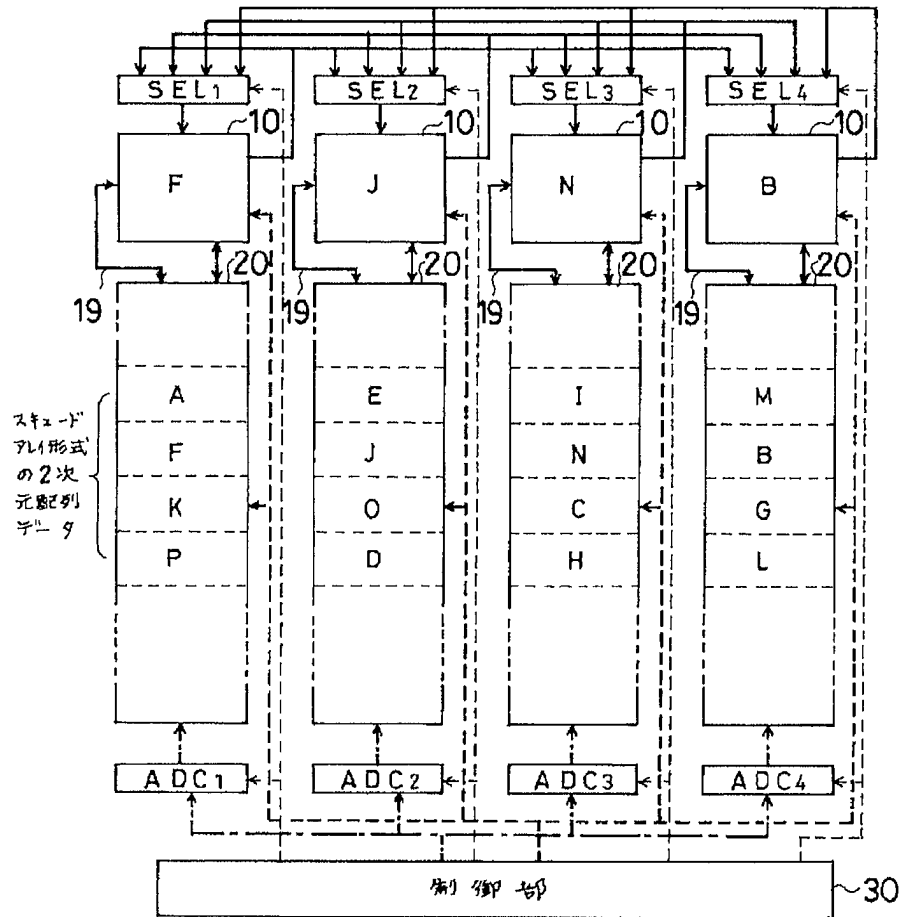


【図9】

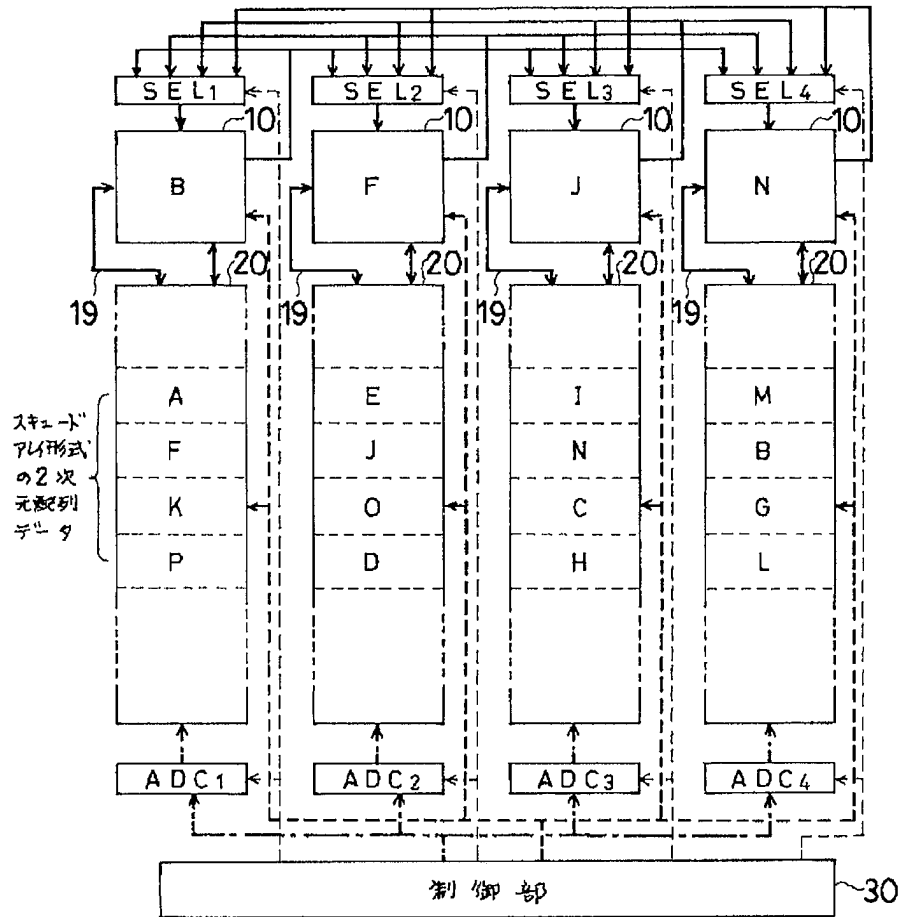




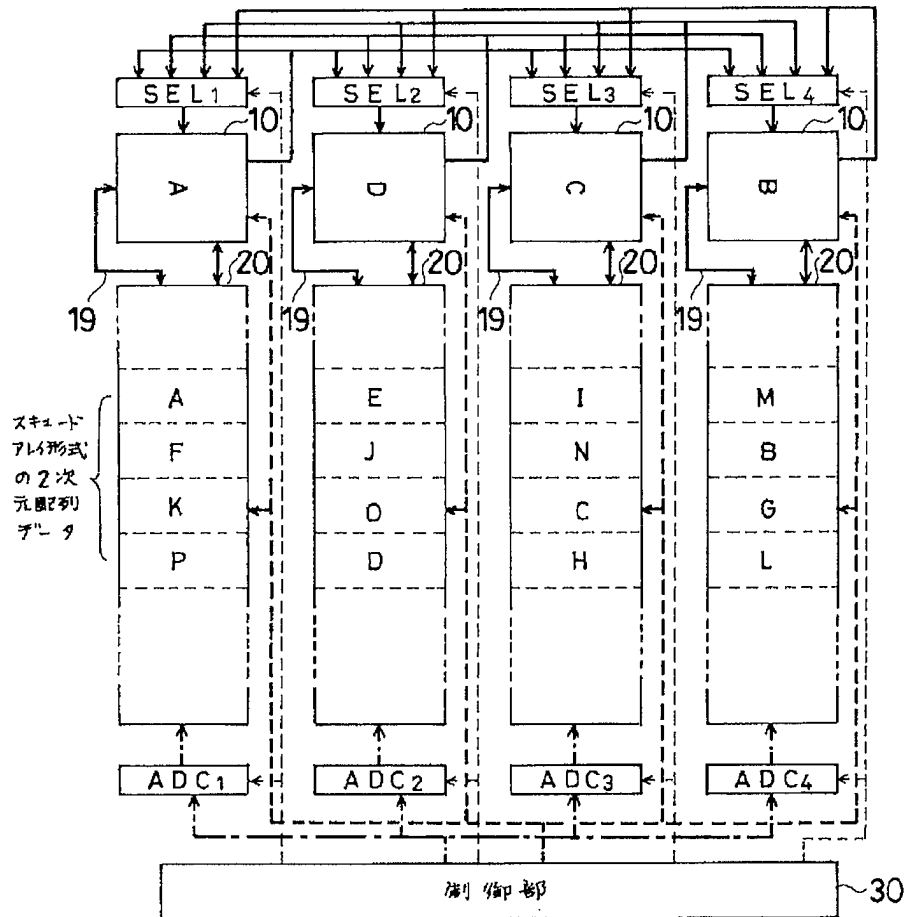
【図10】



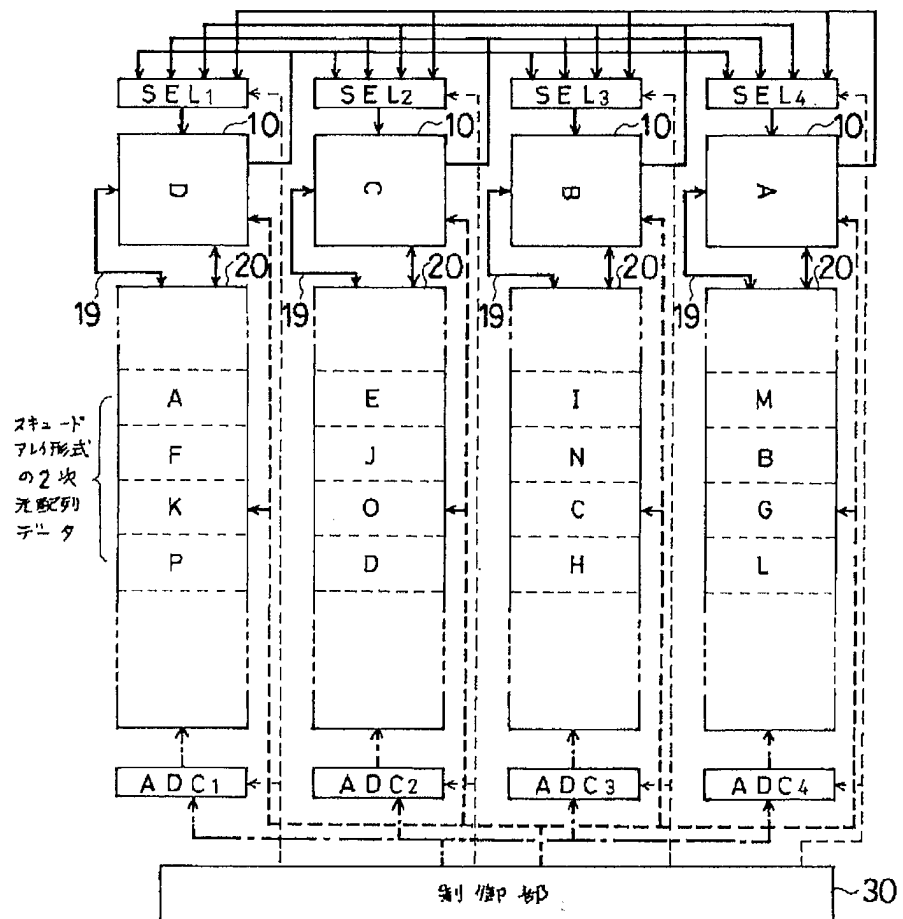
【図11】



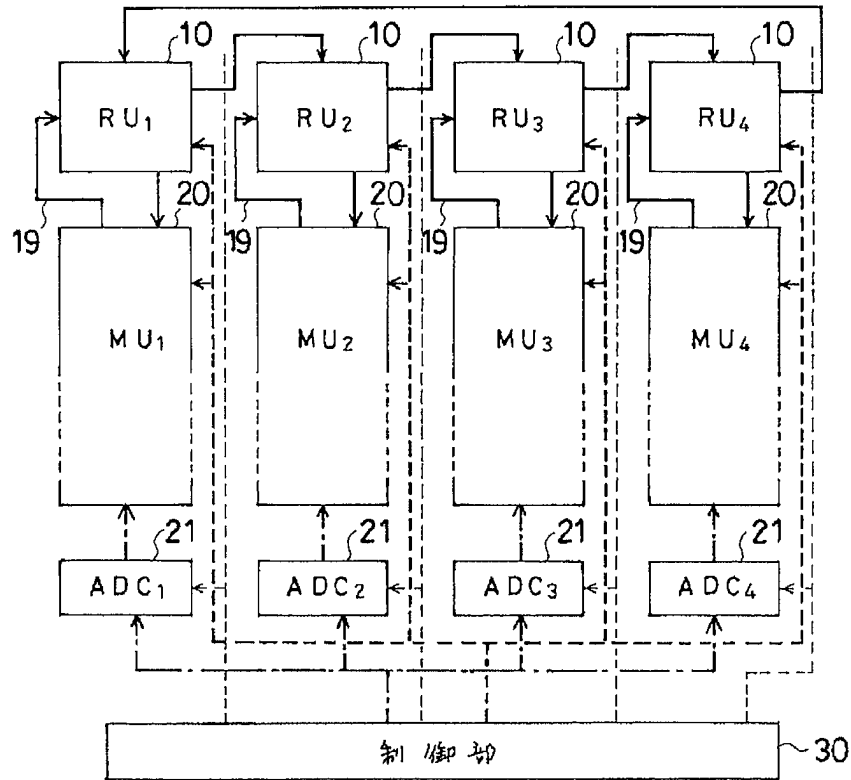
【図12】



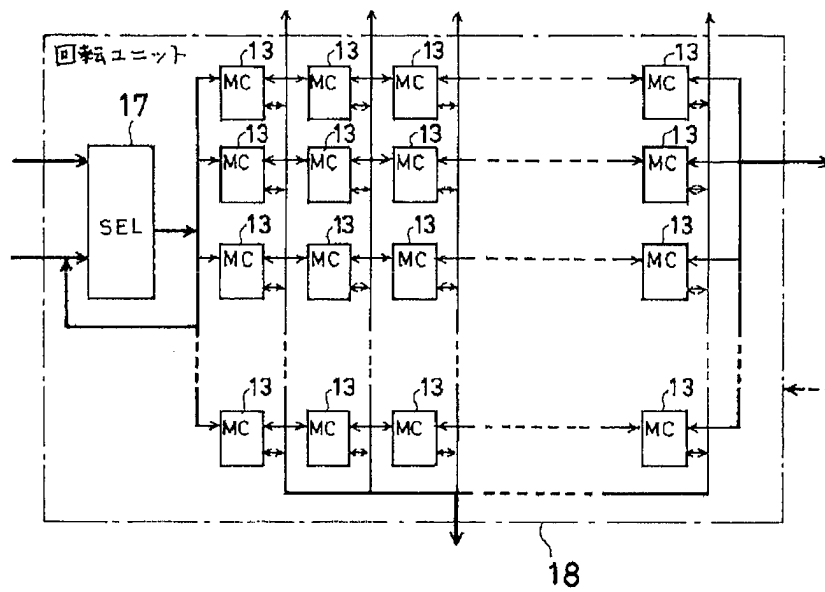
【図13】



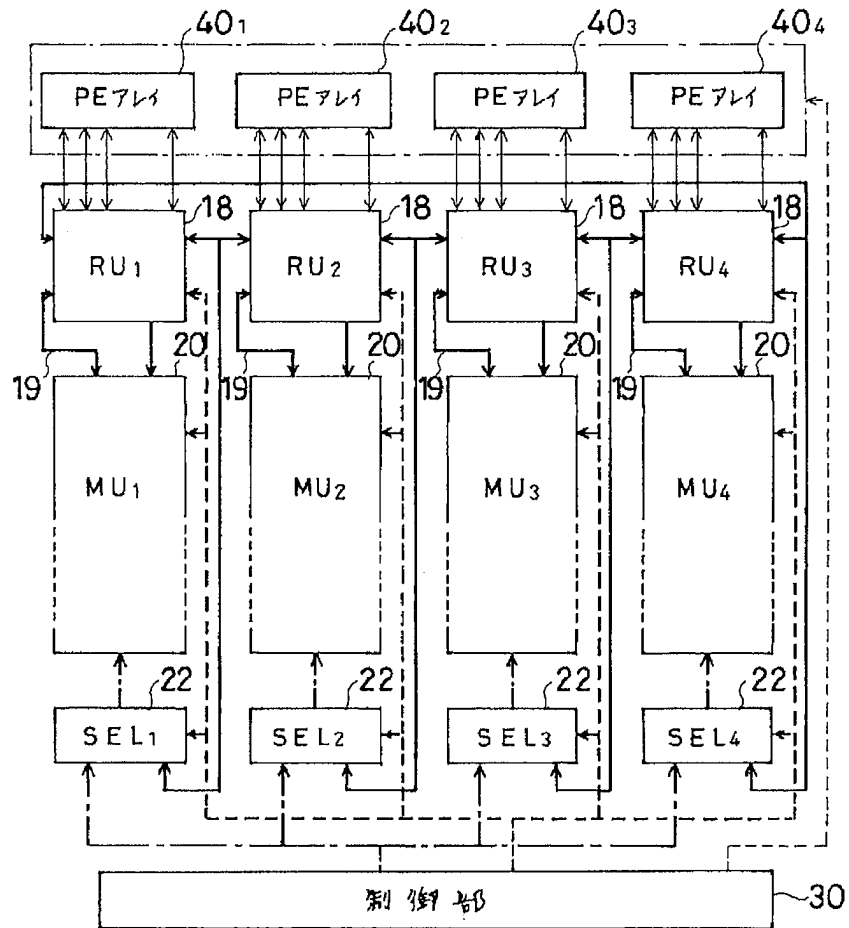
【図15】



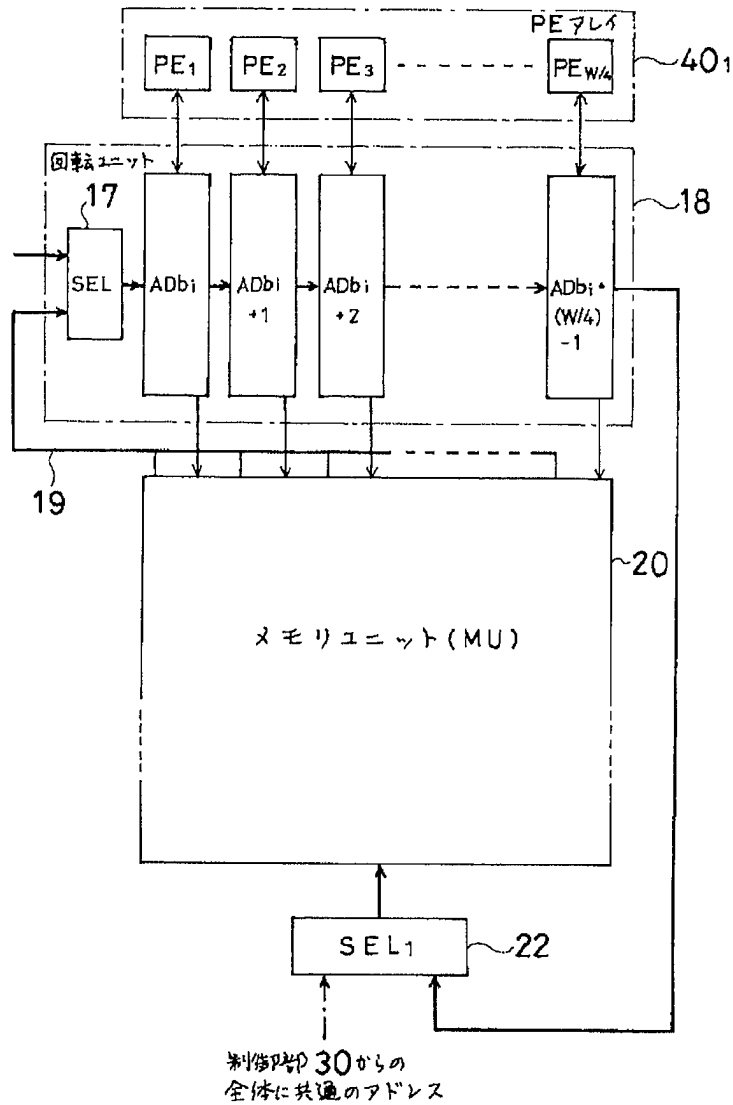
【図17】



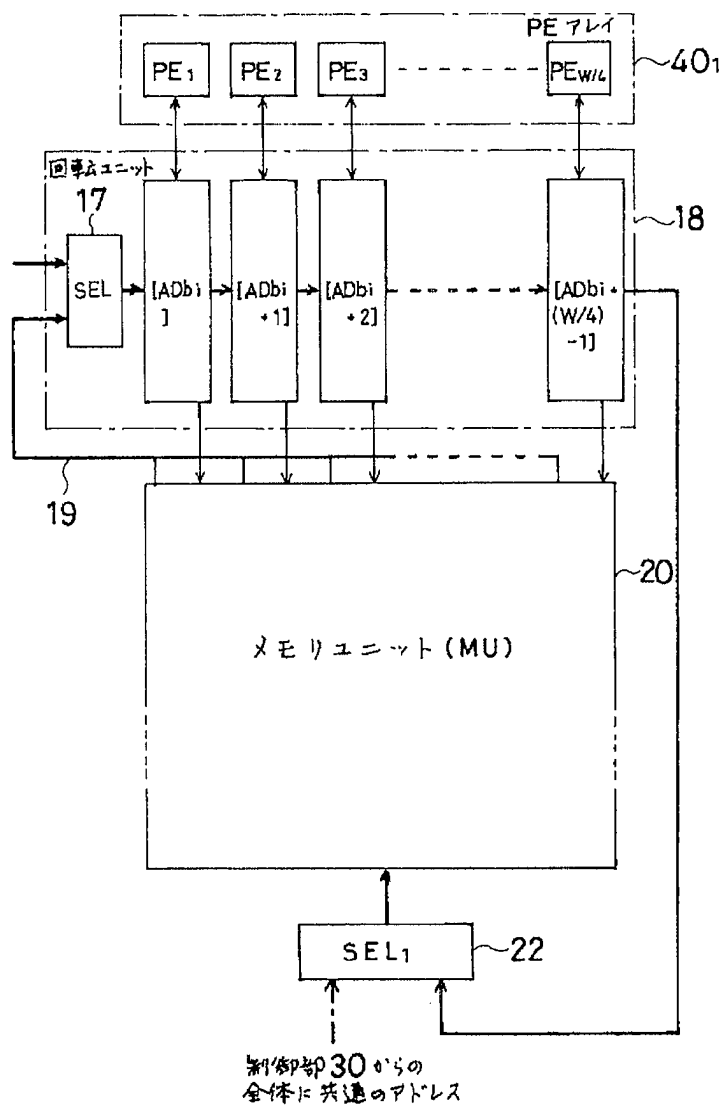
【図16】



【図18】

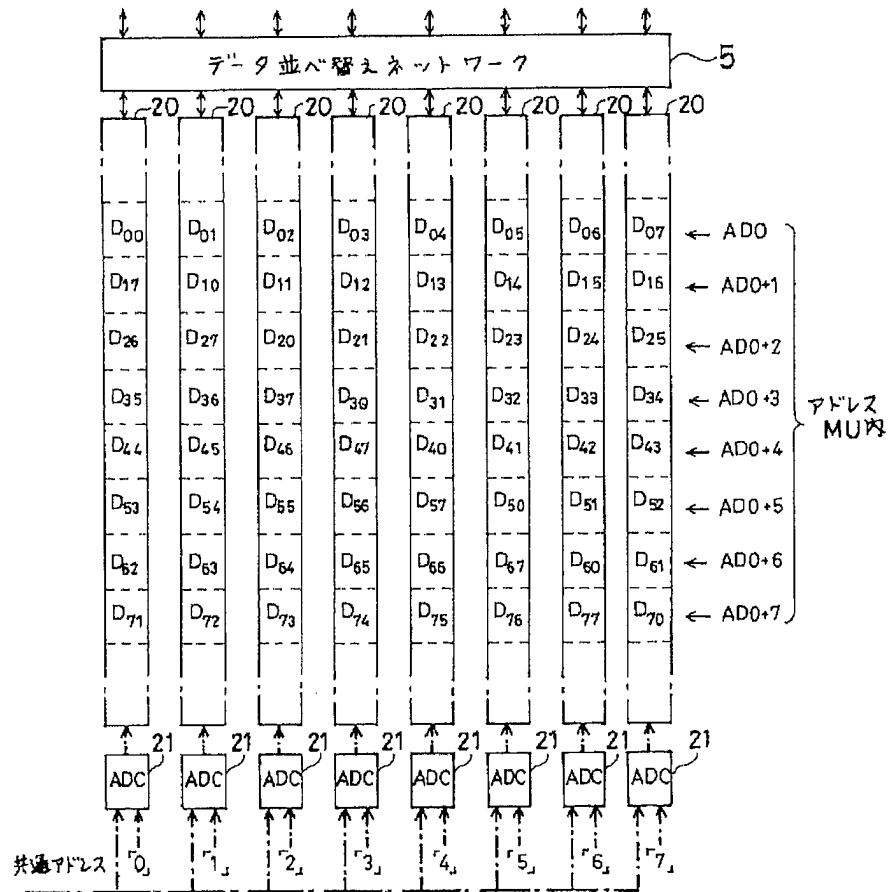


【図19】





【図20】



【図21】

